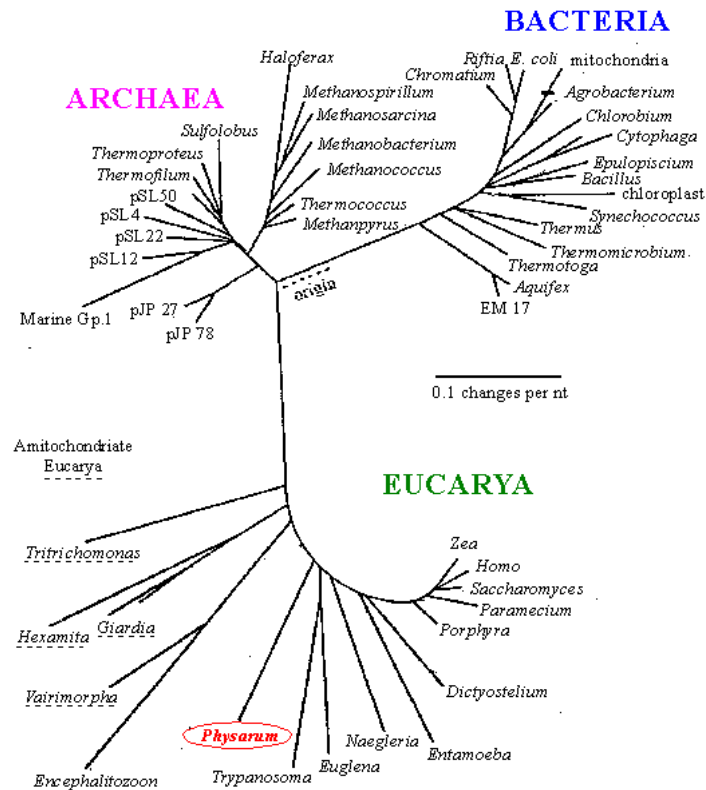

Pairwise Alignment

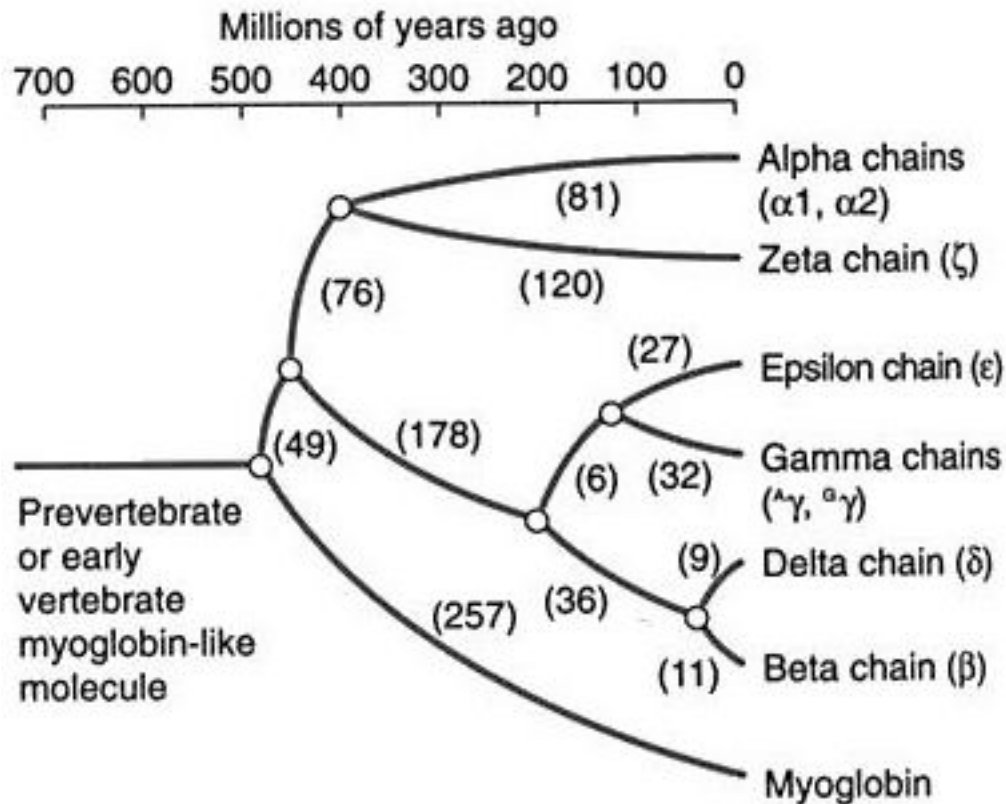
Anders Gorm Pedersen
Molecular Evolution Group
Center for Biological Sequence Analysis

Sequences are related

- Darwin: all organisms are related through descent with modification
- => Sequences are related through descent with modification
- => Similar molecules have similar functions in different organisms

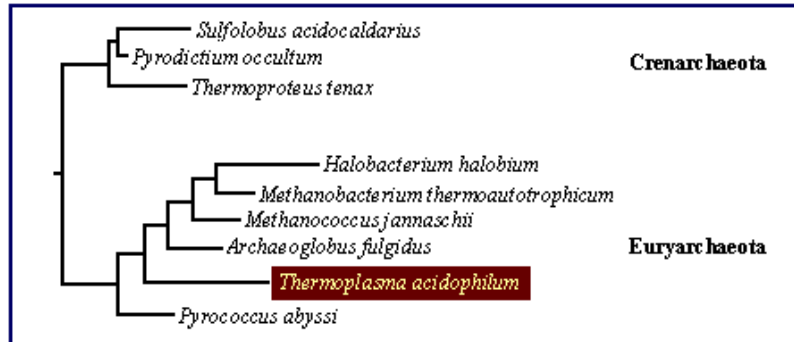


Sequences are related, II



Phylogenetic tree of globin-type proteins found in humans

Why compare sequences?



- Determination of evolutionary relationships

Protein 1: binds oxygen

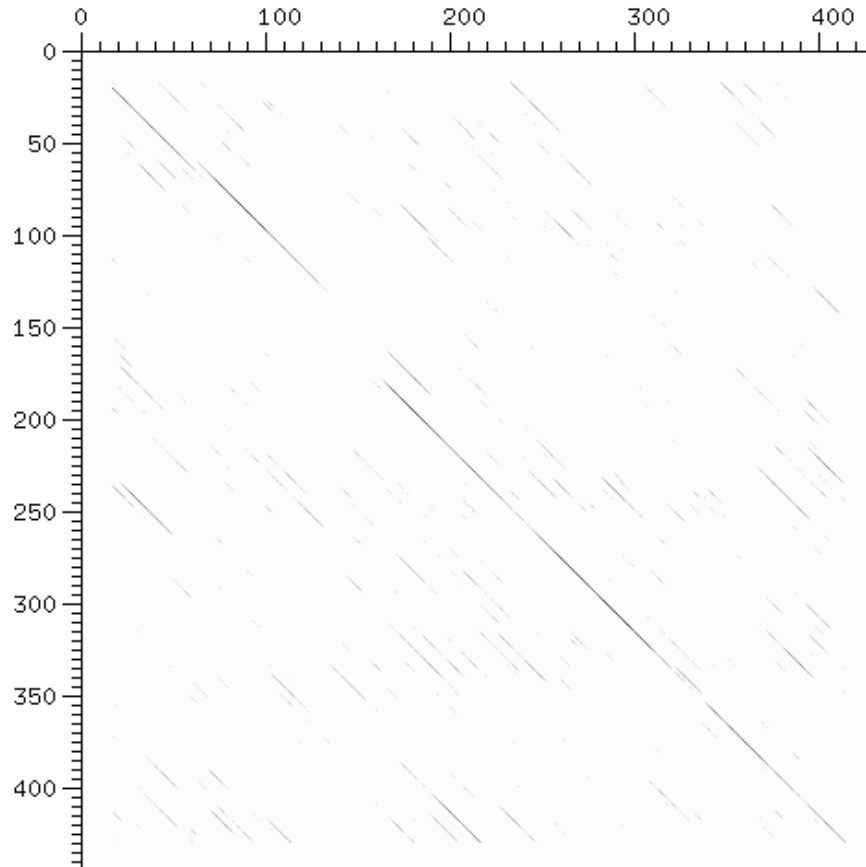


Sequence similarity

Protein 2: binds oxygen ?

- Prediction of protein function and structure (database searches).

Dotplots: visual sequence comparison



1. Place two sequences along axes of plot
2. Place dot at grid points where two sequences have identical residues
3. Diagonals correspond to conserved regions

Global alignment score: 374

	10	20	30	40	50	
alpha	V-LSPADKTNVKAAWGKVGHAHAGEYGAELERMFLSFPTTKTYFPHF-DLS-----HGSA					
	: :. :. : : :. :. :. :. :. :. :. :. :. :. :. :. :.					
beta	VHLTPEEKSAVTALWGKV--NVDEVGGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNP					
	10	20	30	40	50	
	60	70	80	90	100	110
alpha	QVKGHGKKVADALTNAVAHVDDMPNALSALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHL					
	: :. :. :. :. :. :. :. :. :. :. :. :. :. :. :. :. :. :. :. :. :.					
beta	KVKAHGKKVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFRLLGNVLVCVLAHHF					
	60	70	80	90	100	110
	120	130	140			
alpha	PAEFTPAVHASLDKFLASVSTVLTISKYR					
	: :. :. :. :. :. :. :. :. :. :. :. :. :. :. :.					
beta	GKEFTPPVQAAYQKVVAGVANALAHKYH					
	120	130	140			

Pairwise alignment

100.000% identity in 3 aa overlap

SPA

:::

SPA

Percent identity is not a good measure of alignment quality

Global alignment score: 374

	10	20	30	40	50	
alpha	V-LSPADKTNVKAAWGKVGAGHAGEYGAELERMFLSFPTTKTYFPHF-DLS-----HGSA					
	: :. :. : : :. :. :. :. :. :. :. :. :. :. :. :. :.					
beta	VHLTPEEKSAVTALWGKV--NVDEVGGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNP					
	10	20	30	40	50	
	60	70	80	90	100	110
alpha	QVKGHGKKVADALTNAVAHVDDMPNALSALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHL					
	: :. :. :. :. :. :. :. :. :. :. :. :. :. :. :. :. :. :. :. :. :. :.					
beta	KVKAHGKKVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFRLLGNVLCVLAHFF					
	60	70	80	90	100	110
	120	130	140			
alpha	PAEFTPAVHASLDKFLASVSTVLTISKYR					
	: :. :. :. :. :. :. :. :. :. :. :. :. :. :. :. :.					
beta	GKEFTPPVQAAYQKVVAGVANALAHKYH					
	120	130	140			

Alignment scores: match vs. mismatch

Simple scoring scheme (too simple in fact...):

Matching amino acids: 5

Mismatch: 0

Scoring example:

K	A	W	S	A	D	V	
:		:	:	:		:	
K	D	W	S	A	E	V	

$$5+0+5+5+5+0+5 = 25$$

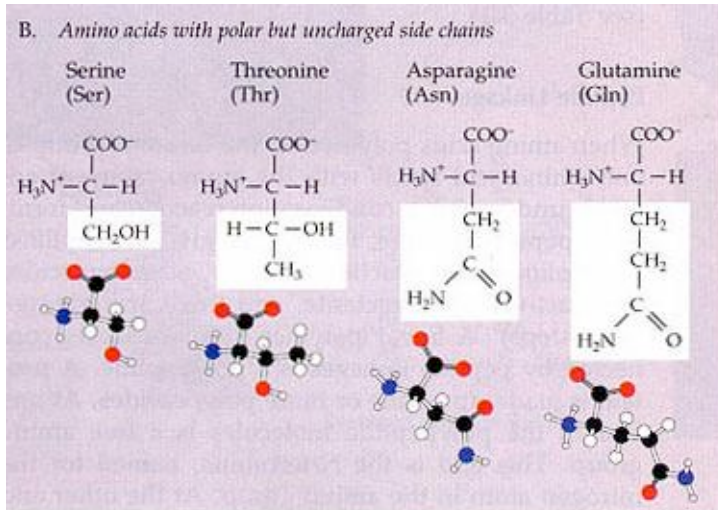
Pairwise alignments: conservative substitutions

43.2% identity;

Global alignment score: 374

	10	20	30	40	50
alpha	V-LSPADKTNVKA	AWGKVG	AHAGEYGA	EALERMFLSF	PTTKTYFP
	:	:	:	:	:
beta	VHLTPEEKSA	VTALWGKV	--NVDE	VGGEALGR	LLVYPWTQ
	:	:	:	:	:
	10	20	30	40	50
	60	70	80	90	100
alpha	QVKG	HGKKV	ADAL	TNAVA	HVDD
	:	:	:	:	:
beta	KVKA	HGKKV	LGAF	SDGL	AHL
	:	:	:	:	:
	60	70	80	90	100
	120	130	140		
alpha	PAEFT	PAVHAS	LDKFL	ASVST	VLTSKYR
	:	:	:	:	:
beta	GKEFT	PPVQA	AYQK	VVAG	VANALAHKYH
	:	:	:	:	:
	120	130	140		

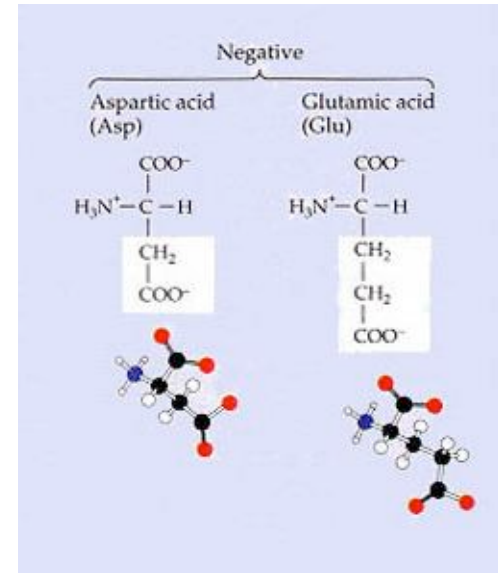
Amino acid properties



Serine (S) and Threonine (T) have similar physicochemical properties

=> Substitution of S/T or E/D occurs relatively often during evolution

=> Substitution of S/T or E/D should result in scores that are only moderately lower than identities



Aspartic acid (D) and Glutamic acid (E) have similar properties

Protein substitution matrices

BLOSUM50 matrix:

A	5																			
R	-2	7																		
N	-1	-1	7																	
D	-2	-2	2	8																
C	-1	-4	-2	-4	13															
Q	-1	1	0	0	-3	7														
E	-1	0	0	2	-3	2	6													
G	0	-3	0	-1	-3	-2	-3	8												
H	-2	0	1	-1	-3	1	0	-2	10											
I	-1	-4	-3	-4	-2	-3	-4	-4	-4	5										
L	-2	-3	-4	-4	-2	-2	-3	-4	-3	2	5									
K	-1	3	0	-1	-3	2	1	-2	0	-3	-3	6								
M	-1	-2	-2	-4	-2	0	-2	-3	-1	2	3	-2	7							
F	-3	-3	-4	-5	-2	-4	-3	-4	-1	0	1	-4	0	8						
P	-1	-3	-2	-1	-4	-1	-1	-2	-2	-3	-4	-1	-3	-4	10					
S	1	-1	1	0	-1	0	-1	0	-1	-3	-3	0	-2	-3	-1	5				
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	2	5			
W	-3	-3	-4	-5	-5	-1	-3	-3	-3	-3	-2	-3	-1	1	-4	-4	-3	15		
Y	-2	-1	-2	-3	-3	-1	-2	-3	2	-1	-1	-2	0	4	-3	-2	-2	2	8	
V	0	-3	-3	-4	-1	-3	-3	-4	-4	4	1	-3	1	-1	-3	-2	0	-3	-1	5
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V

- Positive scores on diagonal (identities)

- Similar residues get higher (positive) scores

- Dissimilar residues get smaller (negative) scores

Pairwise alignments: insertions/deletions

43.2% identity;

Global alignment score: 374

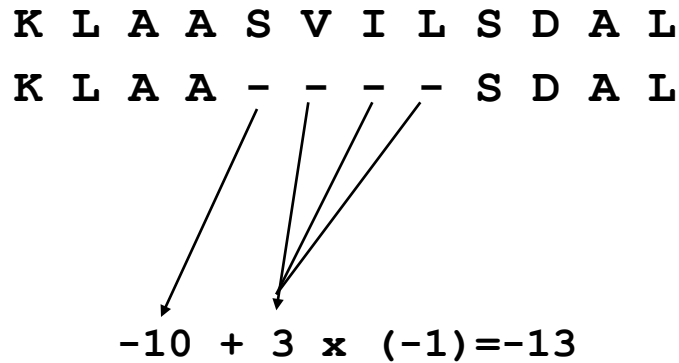
```

      10      20      30      40      50
alpha  V-LSPADKTNVKAAWGKVGAHAGEYGAEALERMFLSFPTTKTYFPHF-DLS-----HGSA
      :  ::  ::  :  :  ::::  ..  :  :::::  ....  :  :  ::  :  :::  :  :
beta   VHLTPEEKSAVTALWGKV--NVDEVGGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNP
      10      20      30      40      50

      60      70      80      90     100     110
alpha  QVKGHGKKVADALTNAVAHVDDMPNALSALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHL
      .....:  .....:  .....:  .....:  .....:  ..  ::  ::
beta   KVKAHGKKVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFRLLGNVLCVLAHFF
      60      70      80      90     100     110

      120     130     140
alpha  PAEFTPAVHASLDKFLASVSTVLTSKYR
      ::::  ::::  :  .....:  ::
beta   GKEFTPPVQAAYQKVVAGVANALAHKYH
      120     130     140
```

Alignment scores: insertions/deletions



Affine gap penalties:

Multiple insertions/deletions may be one evolutionary event =>

Separate penalties for **gap opening** and **gap elongation**

Handout

Compute 4 alignment scores: two different alignments using two different alignment matrices (and the same gap penalty system)

Score 1: Alignment 1 + BLOSUM-50 matrix + gaps

Score 2: Alignment 1 + BLOSUM-Trp matrix + gaps

Score 3: Alignment 2 + BLOSUM-50 matrix + gaps

Score 4: Alignment 2 + BLOSUM-Trp matrix + gaps



Note: fake matrix constructed for pedagogic purposes.

Handout: summary of results

	Alignment 1	Alignment 2
BLOSUM-50	38	51
BLOSUM-Trp	118	91

Protein substitution matrices

A	5																			
R	-2	7																		
N	-1	-1	7																	
D	-2	-2	2	8																
C	-1	-4	-2	-4	13															
Q	-1	1	0	0	-3	7														
E	-1	0	0	2	-3	2	6													
G	0	-3	0	-1	-3	-2	-3	8												
H	-2	0	1	-1	-3	1	0	-2	10											
I	-1	-4	-3	-4	-2	-3	-4	-4	-4	5										
L	-2	-3	-4	-4	-2	-2	-3	-4	-3	2	5									
K	-1	3	0	-1	-3	2	1	-2	0	-3	-3	6								
M	-1	-2	-2	-4	-2	0	-2	-3	-1	2	3	-2	7							
F	-3	-3	-4	-5	-2	-4	-3	-4	-1	0	1	-4	0	8						
P	-1	-3	-2	-1	-4	-1	-1	-2	-2	-3	-4	-1	-3	-4	10					
S	1	-1	1	0	-1	0	-1	0	-1	-3	-3	0	-2	-3	-1	5				
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	2	5			
W	-3	-3	-4	-5	-5	-1	-3	-3	-3	-3	-2	-3	-1	1	-4	-4	-3	15		
Y	-2	-1	-2	-3	-3	-1	-2	-3	2	-1	-1	-2	0	4	-3	-2	-2	2	8	
V	0	-3	-3	-4	-1	-3	-3	-4	-4	4	1	-3	1	-1	-3	-2	0	-3	-1	5
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V

BLOSUM50 matrix:

- Positive scores on diagonal (identities)
- Similar residues get higher (positive) scores
- Dissimilar residues get smaller (negative) scores

Protein substitution matrices: different types

- **Identity matrix**
(match vs. mismatch)
- **Chemical properties matrix**
(use knowledge of physicochemical properties to design matrix)
- ➔ • **Empirical matrices**
(based on observed pair-frequencies in hand-made alignments)
 - PAM series
 - BLOSUM series
 - Gonnet

Estimation of the BLOSUM 62 matrix

- BLOSUM matrices are computed based on gap-free alignments in the so-called BLOCKS database. BLOSUM 62 is computed by comparing sequences that are less than 62% identical. BLOSUM 80 is computed from sequences less than 80% identical, etc.
- All pairs of sequences in a block are compared, and the **observed pair frequencies** (p_{ab}) are noted. For instance: $p_{WW} = 0.0065$, $p_{AL} = 0.0044$, etc.
- Expected pair frequencies** are computed from single amino acid frequencies. For instance: $p_A \times p_L = 0.074 \times 0.099 = 0.0073$
- For each amino acid pair the substitution scores are essentially computed as follows (here λ is a scaling factor, used to obtain integer scores):

$$S(a, b) = \frac{1}{\lambda} \ln \left(\frac{p_{\text{obs}}}{p_{\text{exp}}} \right) = \frac{1}{\lambda} \ln \left(\frac{p_{ab}}{p_a \times p_b} \right)$$

```
ID    FIBRONECTIN_2; BLOCK
COG9_CANFA  GNSAGEPCVFPFIFLGKQYSTCTREGRGDGHLWCATT
COG9_RABIT  GNADGAPCHFPFTFEGRSYTACTTDGRSDGMAWCSTT
FA12_HUMAN  LTVTGEPCHFPPFQYHRQLYHKCTHKGRPGPQWCATT
HGFA_HUMAN  LTEDGRPCRFPPFRYGGRLHACTSEGAHRKWCATTH
MANR_HUMAN  GNANGATCAFPFKFENKQYADCTSGRSDGWLWCGTT
MPRI_MOUSE  ETDDGEPVFPFPIYKGSYDECVLGRAKLWCSKTAN
PB1_PIG     AITSDDKCVFPFIYKGNLYFDCTLHDSTYYWCSVTTY
SFP1_BOVIN  ELPEDEECVFPFVYRNRKHFDCVHGSFLFPWCSLDAD
SFP3_BOVIN  AETKDNKCVFPFIYGNKKYFDCTLHGSFLFLWCSLDAD
SFP4_BOVIN  AVFEGPACAFPTYKGGKYYMCTRKNSVLLWCSLDTE
SP1_HORSE   AATDYAKCAFPFVYRGQTYDRCTTDGSLFRISWCSVT
COG2_CHICK  GNSEGAPCVFPFIFLGNKYDSCTSGARNDDGKLWCAST
COG2_HUMAN  GNSEGAPCVFPFTFLGNKYESCTSGRSDGKMWCAAT
COG2_MOUSE  GNSEGAPCVFPFTFLGNKYESCTSGARNDDGKVCAT
COG2_RABIT  GNSEGAPCVFPFTFLGNKYESCTSGARNDDGKVCAT
COG2_RAT    GNSEGAPCVFPFTFLGNKYESCTSGARNDDGKVCAT
COG9_BOVIN  GNADGKPCVFPFTFQGRYSACTSDGRSDGYRWCATT
COG9_HUMAN  GNADGKPCQFPFIFQGSYSACTTDGRSDGYRWCATT
COG9_MOUSE  GNGEGKPCVFPFIFEGRSYSACTTKGRSDGYRWCATT
COG9_RAT    GNGDGKPCVFPFIFEGHSYSACTTKGRSDGYRWCATT
FINC_BOVIN  GNSNGALCHFPFLYNNHNYTDCTSEGRDNDNMKWCATT
FINC_HUMAN  GNSNGALCHFPFLYNNHNYTDCTSEGRDNDNMKWCATT
FINC_RAT    GNSNGALCHFPFLYNNHNYSDCTSEGRDNDNMKWCATT
MPRI_BOVIN  ETEDGEPCVFPFVFNKGSYEECVVESRARLWCATTAN
MPRI_HUMAN  ETDDGVPCVFPFIFNKGSYEECIIESRAKLWCSTTAD
PA2R_BOVIN  GNAHGTPCMFPFQYNQQWHHECTREGREDNLLWCATT
PA2R_RABIT  GNAHGTPCMFPFQYNHQQWHHECTREGRQDDSLWCATT
```

Estimation of the BLOSUM 62 matrix

- Example: A + L:

$$p_A = 0.074, p_L = 0.099 \Rightarrow p_A \times p_L = 0.0073$$

$$p_{AL} = 0.044$$

$$\begin{aligned} S_{AL} &= \frac{1}{\lambda} \ln \left(\frac{p_{AL}}{p_A \times p_L} \right) \\ &= \frac{1}{0.347} \ln \left(\frac{0.0044}{0.0073} \right) \\ &= \frac{-0.51}{0.347} \\ &= -1.46 \\ &\approx -1 \end{aligned}$$

Pairwise alignment

Optimal alignment:

alignment having the highest possible score given a substitution matrix and a set of gap penalties

So:

best alignment can be found by exhaustively searching all possible alignments, scoring each of them and choosing the one with the highest score?

The problem:

How many possible alignments are there?

ACG	AC-G	--ACG	-A-CG
ACG	ACG-	AC-G-	A-CG-
-ACG	AC-G	--ACG	...
ACG-	A-CG	A-CG-	
-ACG	AC-G	--ACG	
AC-G	-ACG	AC--G	
-ACG	ACG-	--ACG	
A-CG	AC-G	A-C-G	
A-CG	ACG-	--ACG	
ACG-	A-CG	A--CG	
A-CG	ACG-	-A-CG	
AC-G	-ACG	ACG--	
A-CG	--ACG	-A-CG	
-ACG	ACG--	AC-G-	

How many possible alignments are there?

Length of sequences: $n_1 = n_2$	Number of possible alignments
2	13
3	63
4	321
5	1683
10	8,097,453
20	2.61×10^{14}
100	2.05×10^{75}
300	1.53×10^{228}

Pairwise alignment: the problem

The number of possible pairwise alignments increases explosively with the length of the sequences:

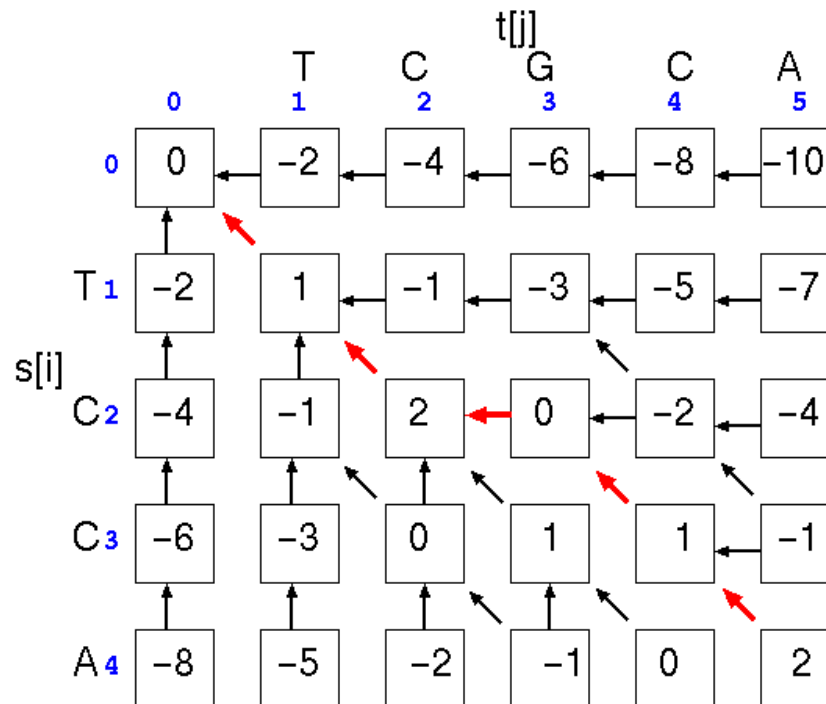
$$N = \sum_{k=0}^{n_2} \binom{n_1 + k}{n_2} \binom{n_2}{k}$$

Two protein sequences of length 300 amino acids can be aligned in approximately 10^{228} different ways

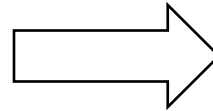
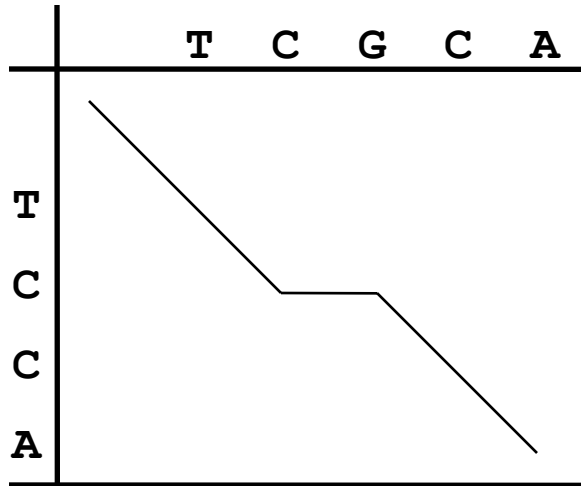
Time needed to test all possibilities much larger than the entire lifetime of the universe.

Pairwise alignment: the solution

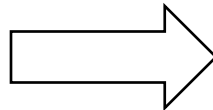
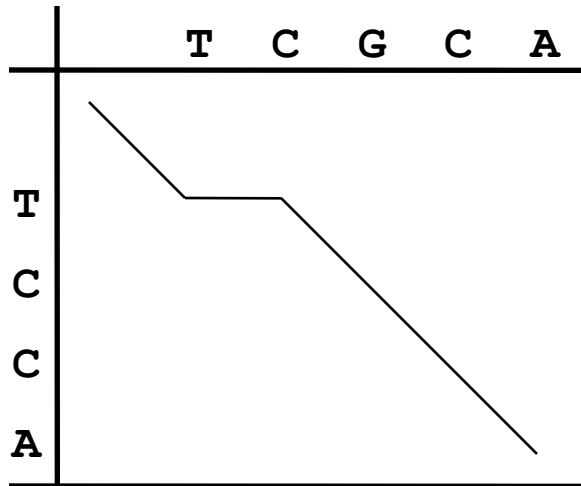
"Dynamic programming"
(the Needleman-Wunsch algorithm)



Alignment depicted as path in matrix



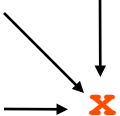
TCGCA
TC-CA



TCGCA
T-CCA

Dynamic programming: computation of scores

	T	C	G	C	A
T					
C					
C					
A					



The diagram shows a 5x5 matrix with columns labeled T, C, G, C, A and rows labeled T, C, C, A. The cell at row 2, column 4 (C, C) is marked with a red 'x'. Three arrows point to this cell: one from the cell above (T, C), one from the cell to the left (C, C), and one from the cell diagonally up and to the left (T, G).

Any given point in matrix can only be reached from three possible previous positions (you cannot “align backwards”).

=> Best scoring alignment ending in any given point in the matrix can be found by choosing the highest scoring of the three possibilities.

Dynamic programming: computation of scores

	T	C	G	C	A
T					
C		x			
C					
A					

Any given point in matrix can only be reached from three possible positions (you cannot “align backwards”).

=> Best scoring alignment ending in any given point in the matrix can be found by choosing the highest scoring of the three possibilities.

$$\text{score}(x,y) = \max \left\{ \begin{array}{l} \text{score}(x,y-1) - \text{gap-penalty} \\ \text{score}(x-1,y) \\ \text{score}(x-1,y-1) \end{array} \right.$$

Dynamic programming: computation of scores

	T	C	G	C	A
T					
C					
C					
A					

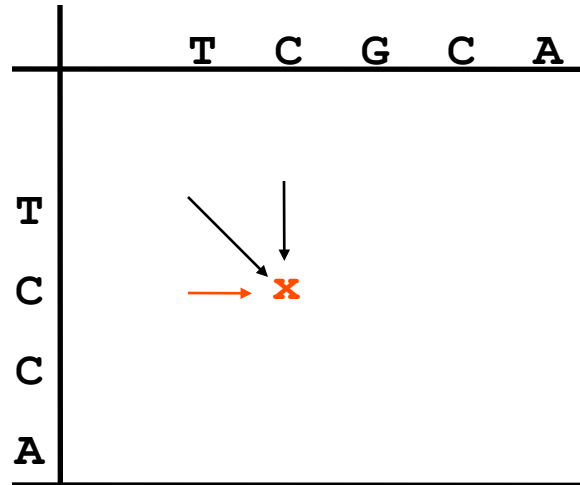
Any given point in matrix can only be reached from three possible positions (you cannot “align backwards”).

=> Best scoring alignment ending in any given point in the matrix can be found by choosing the highest scoring of the three possibilities.

$$\text{score}(x,y) = \max \left\{ \begin{array}{l} \text{score}(x,y-1) - \text{gap-penalty} \\ \text{score}(x-1,y-1) + \text{substitution-score}(x,y) \end{array} \right.$$

Dynamic programming: computation of scores

	T	C	G	C	A
T					
C					
C					
A					



Any given point in matrix can only be reached from three possible positions (you cannot “align backwards”).

=> Best scoring alignment ending in any given point in the matrix can be found by choosing the highest scoring of the three possibilities.

$$\text{score}(x,y) = \max \begin{cases} \text{score}(x,y-1) - \text{gap-penalty} \\ \text{score}(x-1,y-1) + \text{substitution-score}(x,y) \\ \text{score}(x-1,y) - \text{gap-penalty} \end{cases}$$

Dynamic programming: computation of scores

	T	C	G	C	A
T					
C					
C					
A					

Any given point in matrix can only be reached from three possible positions (you cannot “align backwards”).

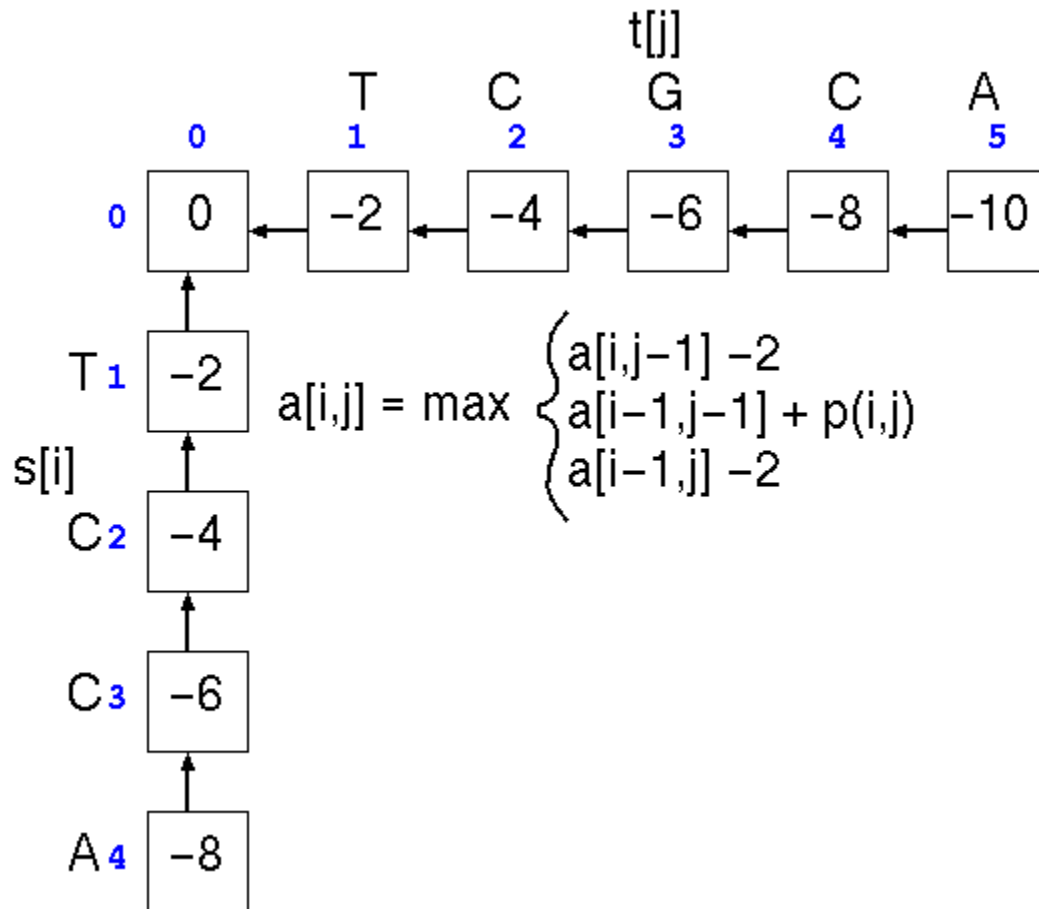
=> Best scoring alignment ending in any given point in the matrix can be found by choosing the highest scoring of the three possibilities.

Each new score is found by choosing the maximum of three possibilities.
For each square in matrix: keep track of where best score came from.

Fill in scores one row at a time, starting in upper left corner of matrix, ending in lower right corner.

$$\text{score}(x,y) = \max \begin{cases} \text{score}(x,y-1) - \text{gap-penalty} \\ \text{score}(x-1,y-1) + \text{substitution-score}(x,y) \\ \text{score}(x-1,y) - \text{gap-penalty} \end{cases}$$

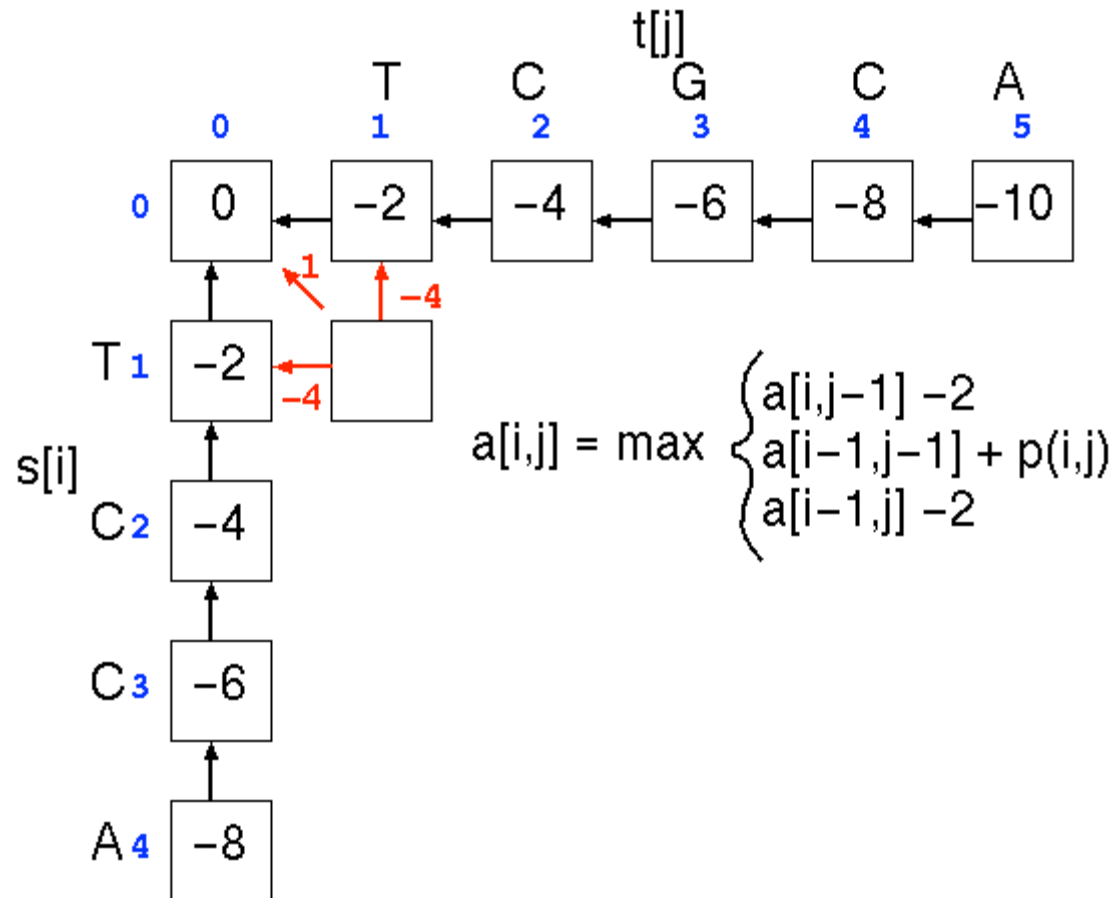
Dynamic programming: example



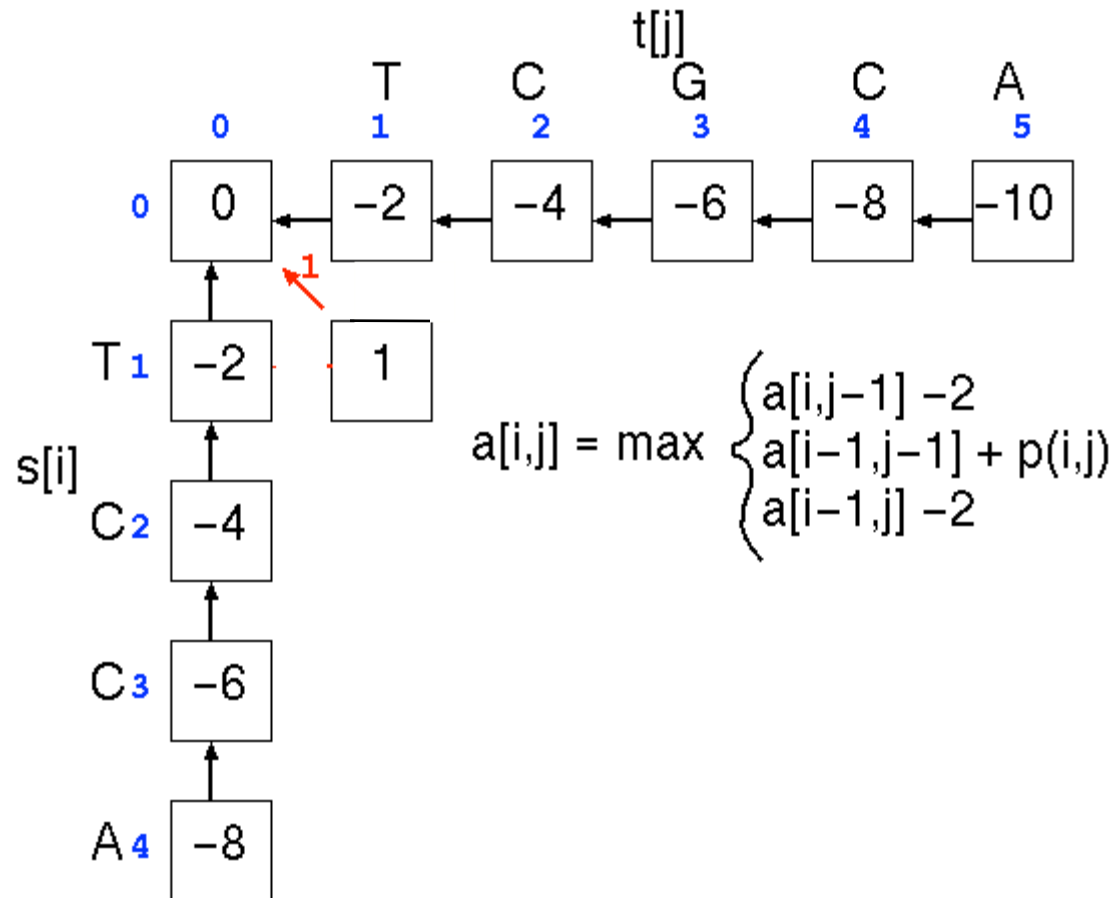
	A	C	G	T
A	1	-1	-1	-1
C	-1	1	-1	-1
G	-1	-1	1	-1
T	-1	-1	-1	1

Gaps: -2

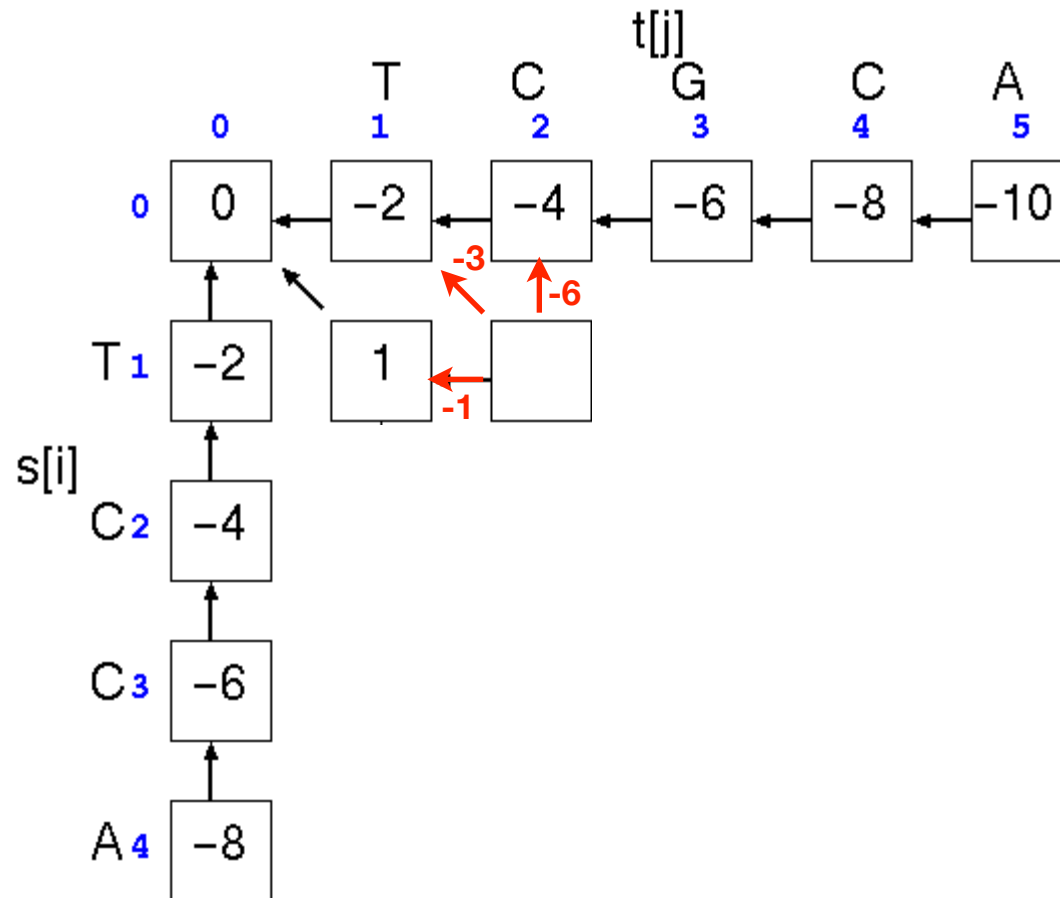
Dynamic programming: example



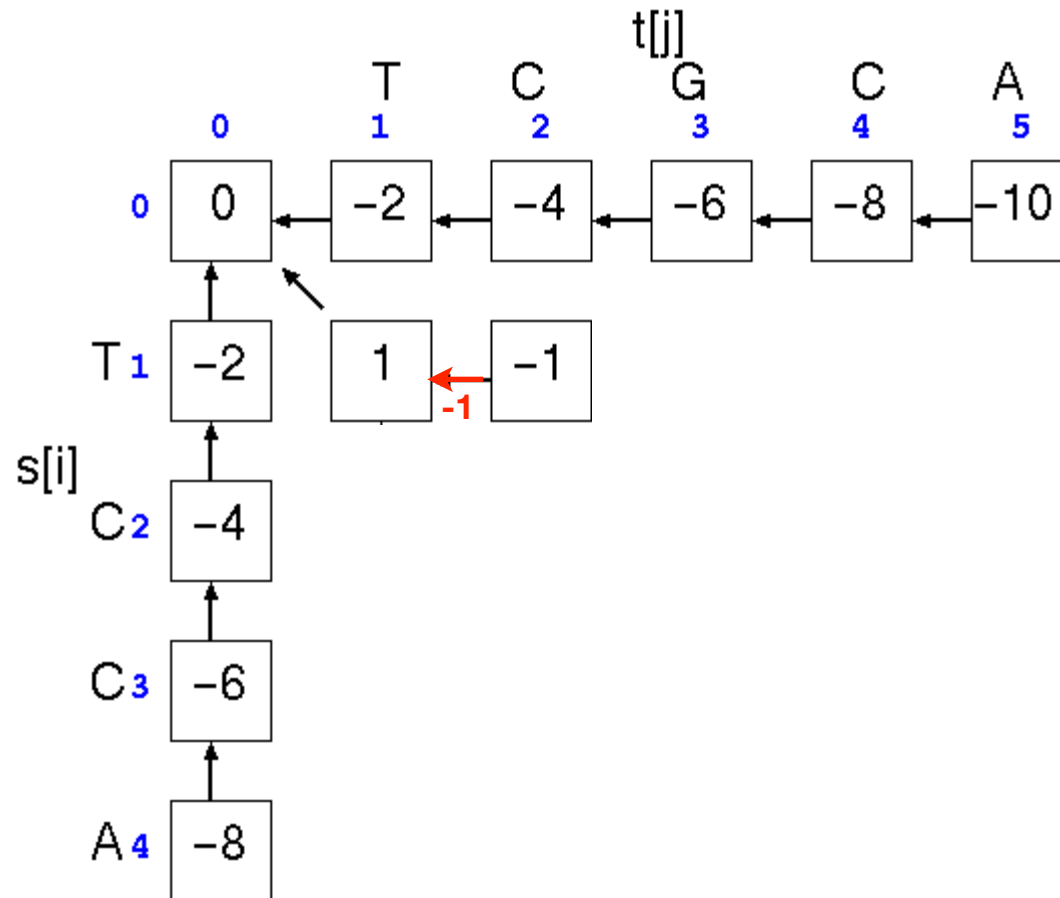
Dynamic programming: example



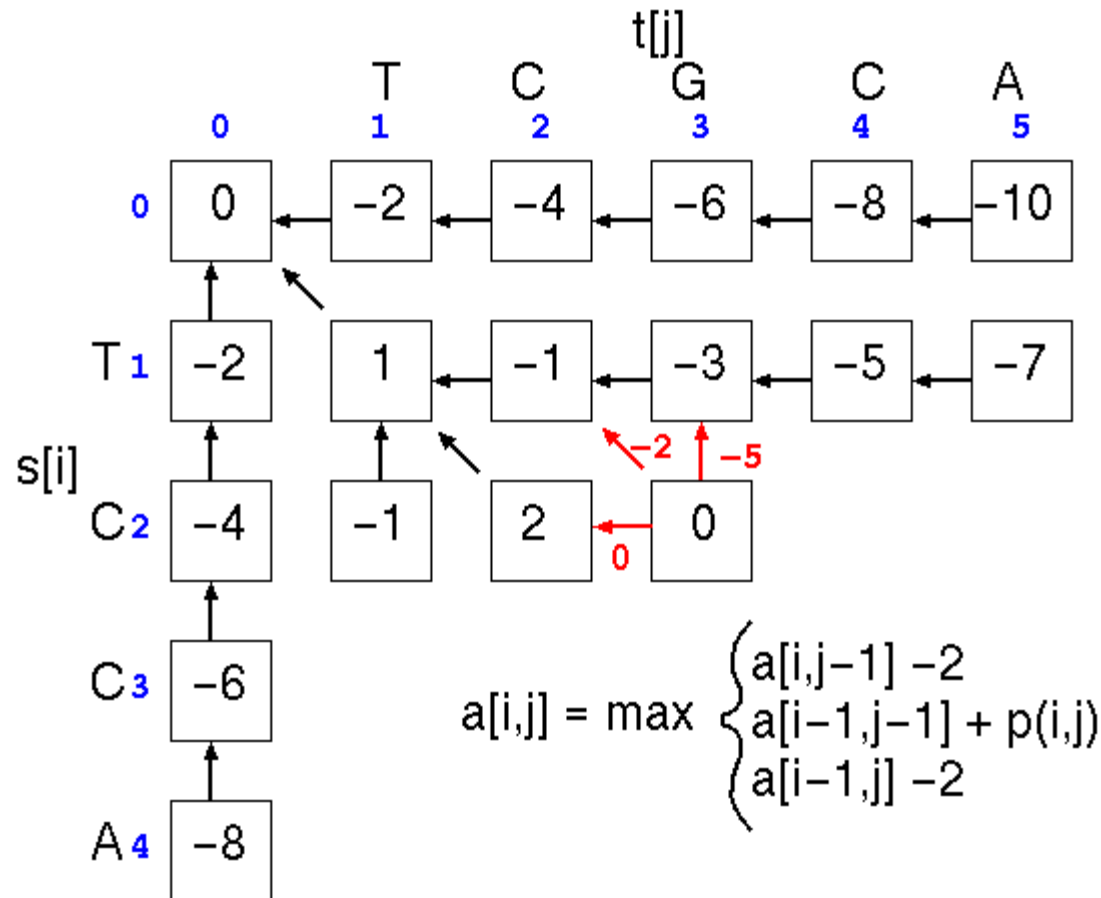
Dynamic programming: example



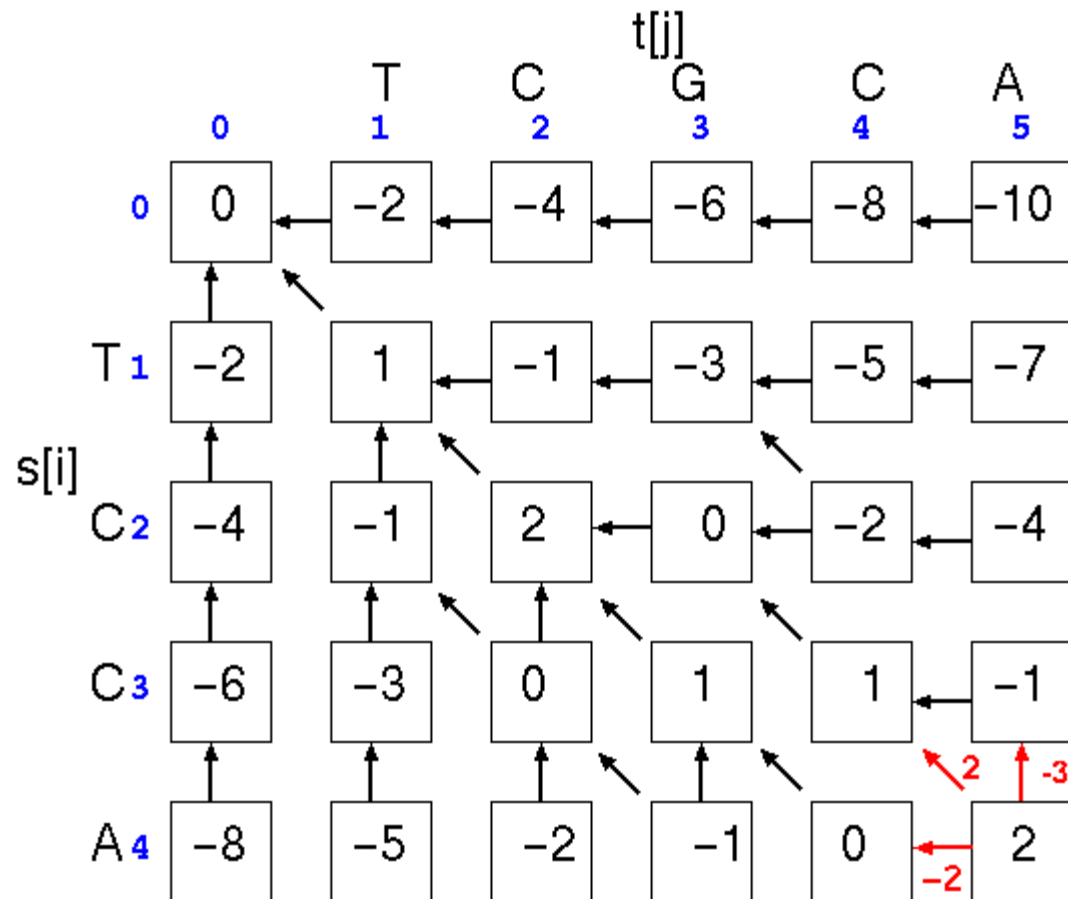
Dynamic programming: example



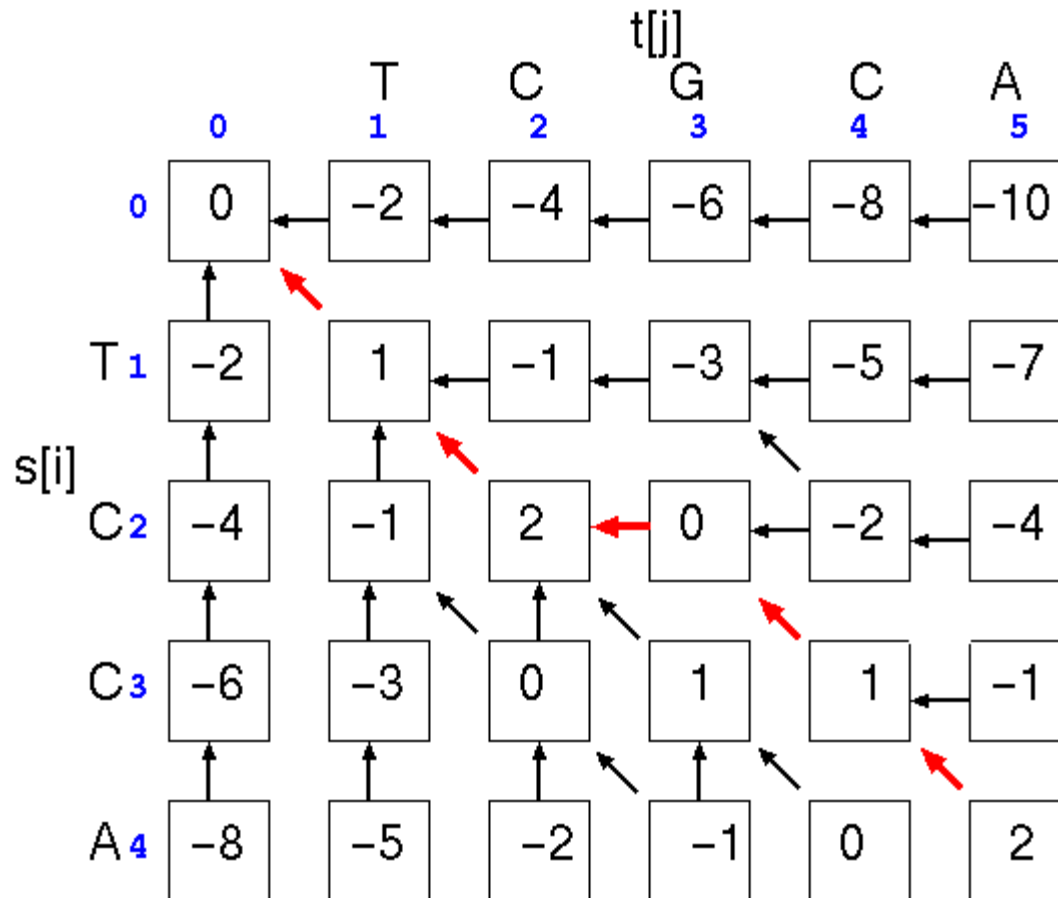
Dynamic programming: example



Dynamic programming: example



Dynamic programming: example



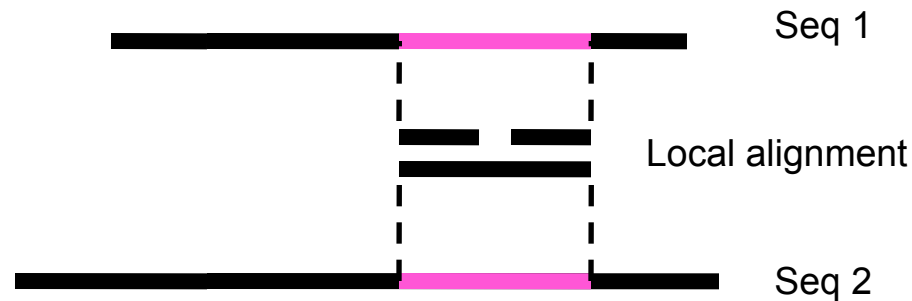
$$\begin{array}{cccccc}
 & \text{T} & \text{C} & \text{G} & \text{C} & \text{A} \\
 & \vdots & \vdots & & \vdots & \vdots \\
 & \text{T} & \text{C} & - & \text{C} & \text{A} \\
 \hline
 & 1 & +1 & -2 & +1 & +1 & = & \underline{2}
 \end{array}$$

Global versus local alignments

Global alignment: align full length of both sequences.
(The “Needleman-Wunsch” algorithm).



Local alignment: find best partial alignment of two sequences
(the “Smith-Waterman” algorithm).



Local alignment overview

- The recursive formula is changed by adding a fourth possibility: zero. This means local alignment scores are never negative.

$$\text{score}(x,y) = \max \begin{cases} \text{score}(x,y-1) - \text{gap-penalty} \\ \text{score}(x-1,y-1) + \text{substitution-score}(x,y) \\ \text{score}(x-1,y) - \text{gap-penalty} \\ 0 \end{cases}$$

- Trace-back is started at the highest value rather than in lower right corner
- Trace-back is stopped as soon as a zero is encountered

Local alignment: example

		H	E	A	G	A	W	G	H	E	E
	0	0	0	0	0	0	0	0	0	0	0
P	0	0	0	0	0	0	0	0	0	0	0
A	0	0	0	5	0	5	0	0	0	0	0
W	0	0	0	0	2	0	20	12	4	0	0
H	0	10	2	0	0	0	12	18	22	14	6
E	0	2	16	8	0	0	4	10	18	28	20
A	0	0	8	21	13	5	0	4	10	20	27
E	0	0	6	13	18	12	4	0	4	16	26

AWGHE

AW-HE

Substitution matrices and sequence similarity

- Substitution matrices come as series of matrices calculated for different degrees of sequence similarity (different evolutionary distances).
- "Hard" matrices are designed for similar sequences
 - Hard matrices are designated by high numbers in the BLOSUM series (e.g., BLOSUM80)
 - Hard matrices yield short, highly conserved alignments
- "Soft" matrices are designed for less similar sequences
 - Soft matrices have low BLOSUM values (45)
 - Soft matrices yield longer, less well conserved alignments

Alignments: things to keep in mind

“Optimal alignment” means “having the highest possible score, given substitution matrix and set of gap penalties”.

This is NOT necessarily the biologically most meaningful alignment.

Specifically, the underlying assumptions are often wrong: substitutions are not equally frequent at all positions, affine gap penalties do not model insertion/deletion well, etc.

Pairwise alignment programs always produce an alignment - even when it does not make sense to align sequences.